Parallel MRI
k-Space Techniques

ESMRMB Workshop in Parallel Imaging

June 2008, Lisbon, Portugal

Felix Breuer

Magnetic Resonance Bavaria, Würzburg Germany

Overview

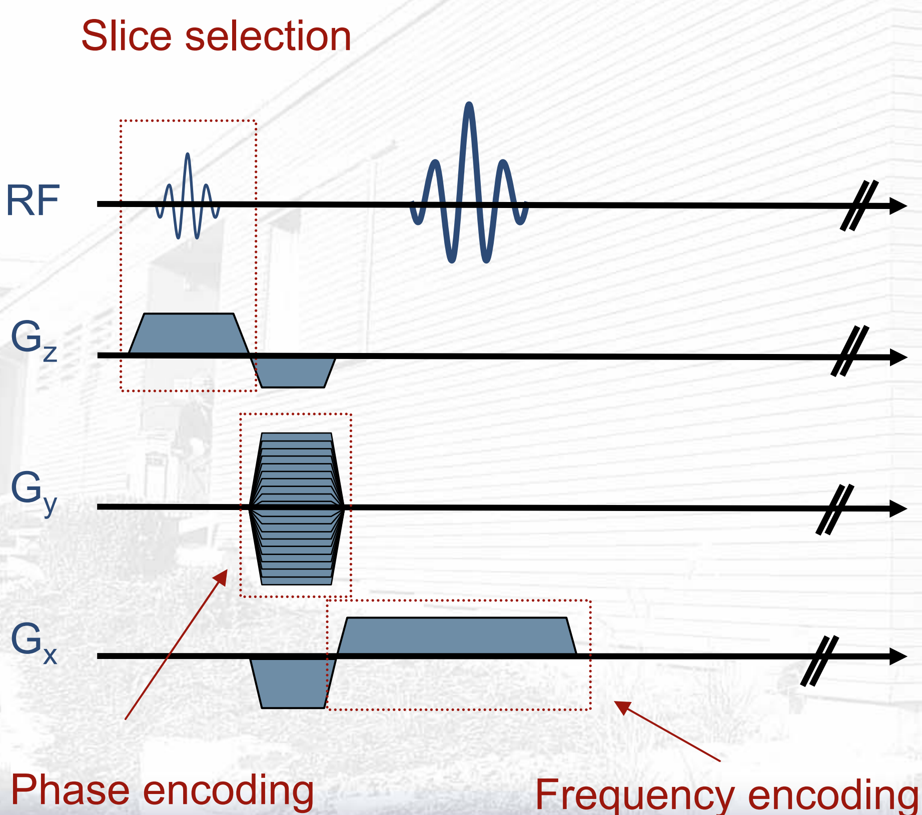
- **Spatial Encoding in MRI**
- **How does parallel MRI work?**
  - Image space
  - K-space
- **What kind of k-space pMRI methods are available and how do they work?**
  - SMASH
  - Generalized SMASH
  - GRAPPA
- **What are the advantages, disadvantages?**

Spatial encoding with gradients

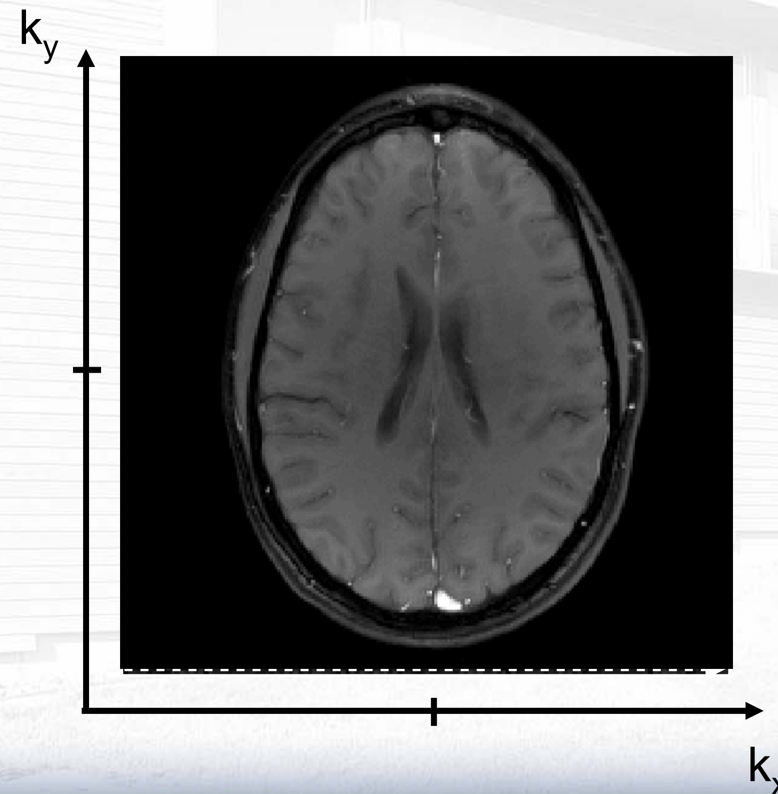
k-space principle:

$$k_x = \int_0^T G_x dt \qquad k_x = \int_0^T G_y dt$$

MR Pulse Sequence

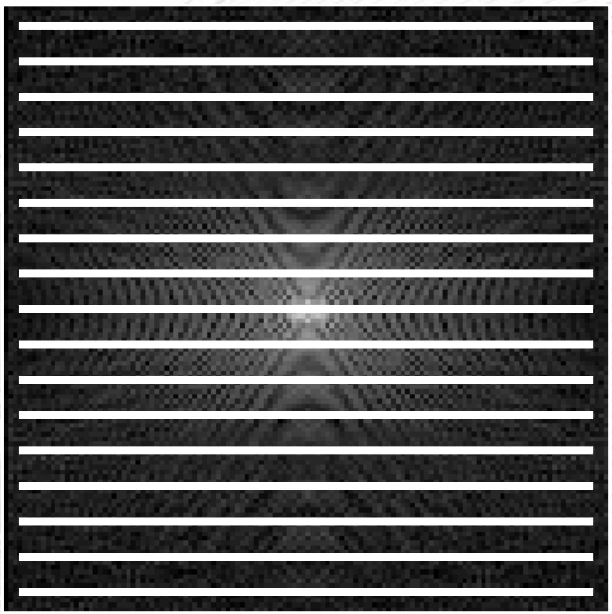


k-Space

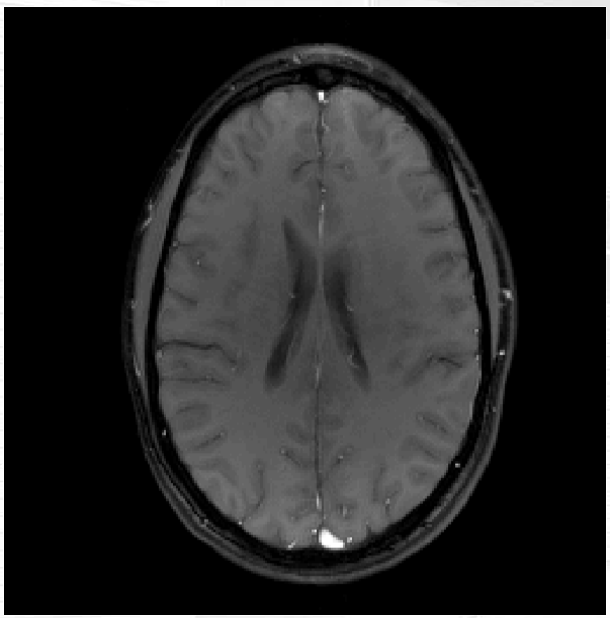


Nyquist Criterion

$$\Delta k_y = \frac{2\pi}{FOV_y}$$



$\} \; \Delta k_y$

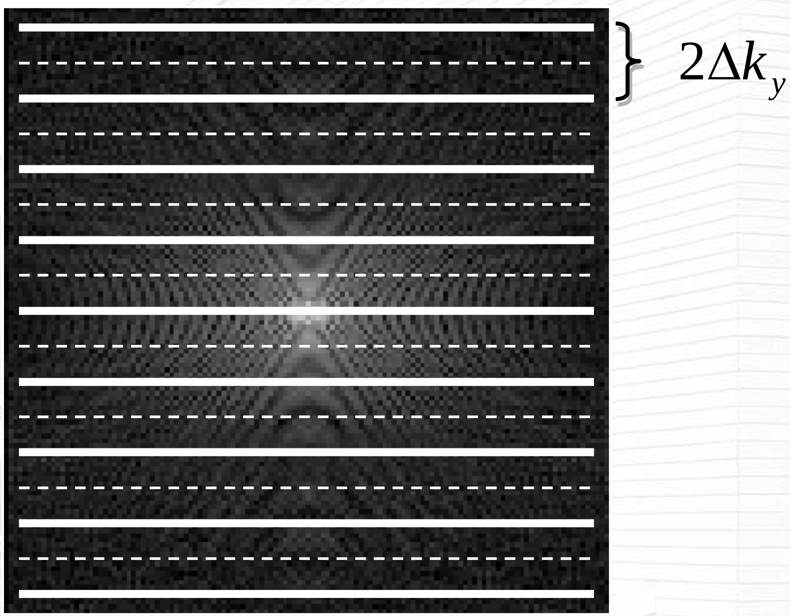


FOV

K-Space

Image - Space

Nyquist Criterion



K-Space

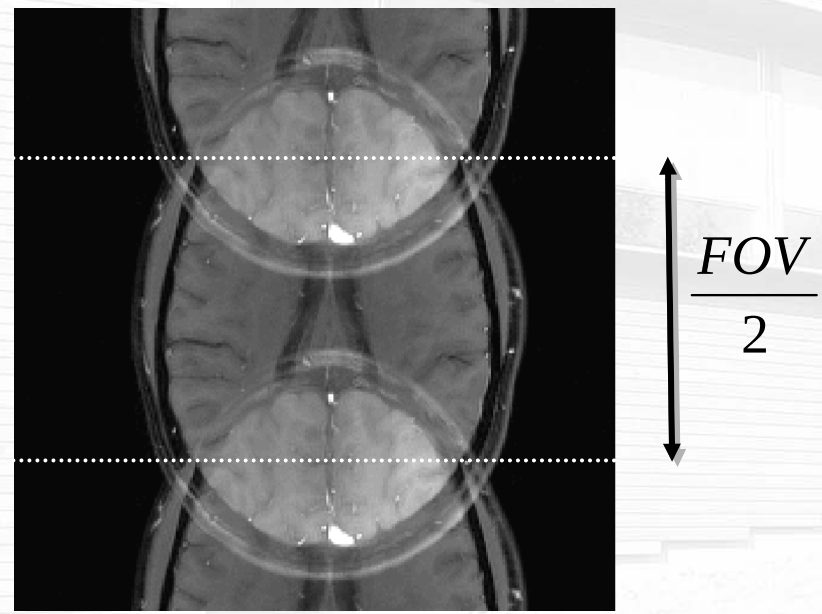
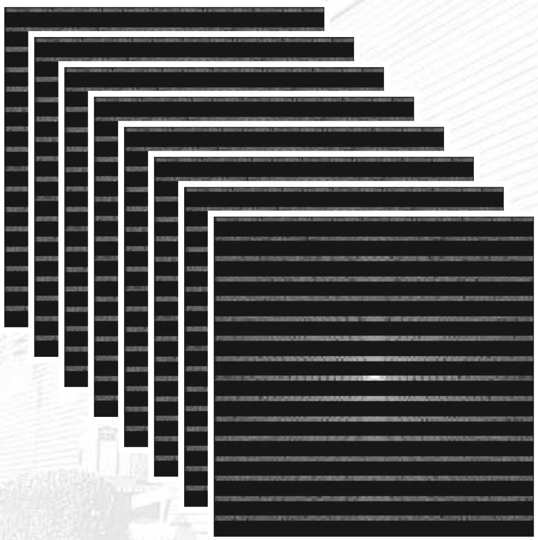


Image - Space

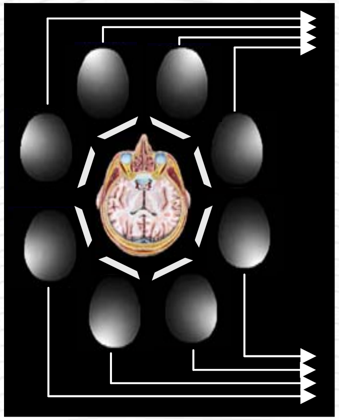
Spatial Encoding with coils

Reduced multi-coil dataset with missing k-space data



&

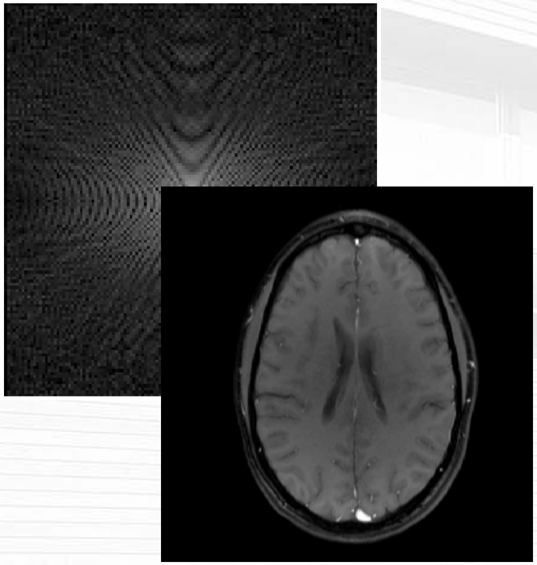
Coil sensitivity information



=

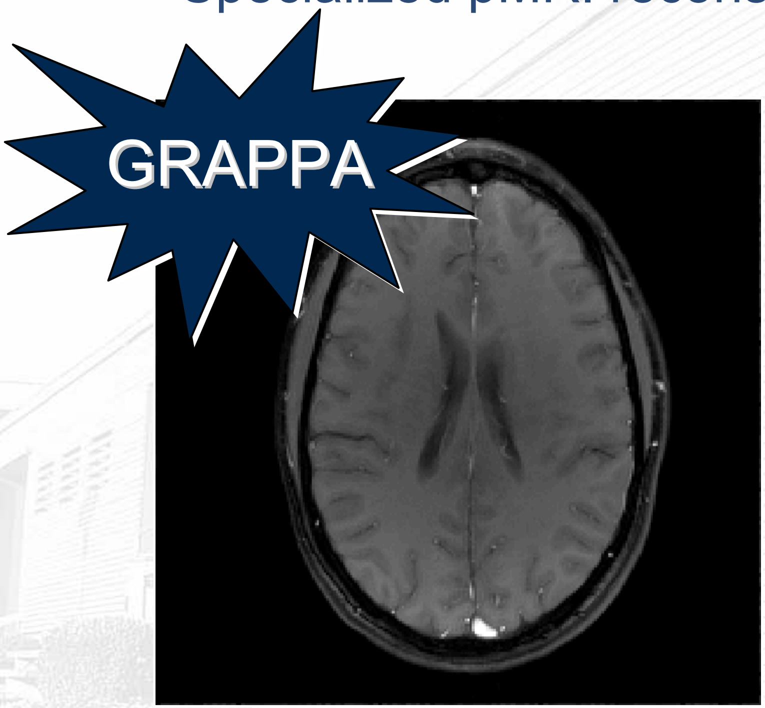
fully sampled k-space/

Alias-free image



Parallel MRI

- Signal detection with multiple receiver coils (up to 32)
- Data reduction in the phase encoding direction (Image acceleration)
- Specialized pMRI reconstruction algorithm (SENSE / GRAPPA)



K-Space

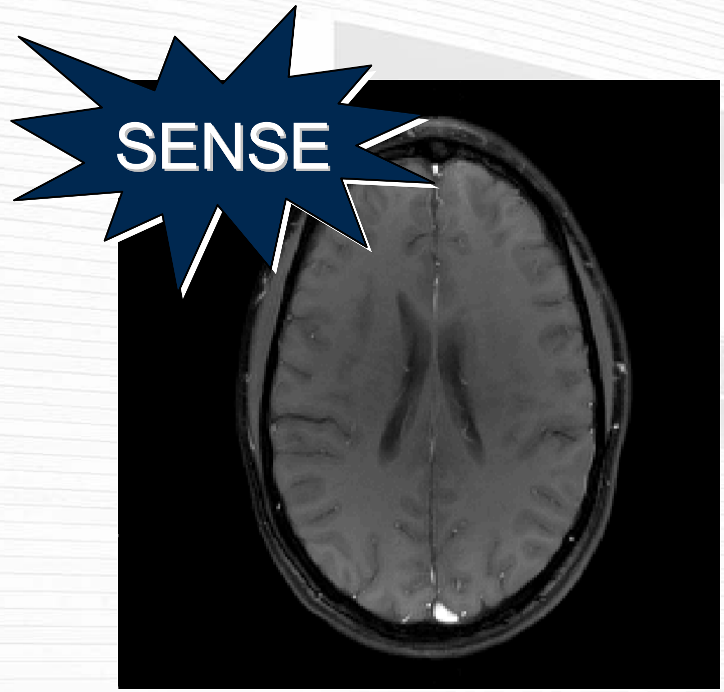
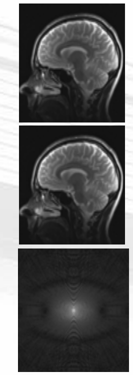


Image - Space

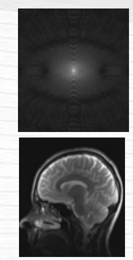
Both strategies rely on extra knowledge of sensitivity information!

Milestones

- **Kelton, Magin & Wright (1989)**
- **Ra & Rim (1990)**
- **Carlson, Minemura (1993)**

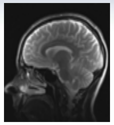


- **Sodickson & Manning SMASH (1997)**
- **Pruessmann, Weiger, et al SENSE (1998)**

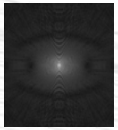


Milestones

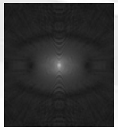
- **Griswold, et al PILS (2000)**



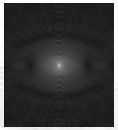
- **Kyriakos, et al SPACE-RIP (2000)**



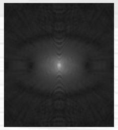
- **Griswold, et al GRAPPA (2001)**



- **Bydder, et al G-SMASH (2002)**



- **Yeh, et al PARS (2005)**



pMRI reconstructions in k-space

SMASH (sensitivity maps)

Generalized SMASH (sensitivity maps)

GRAPPA (calibration-scan)

Review: Spatial encoding in MRI

k-space principle:

$$S(k_y, k_x) = \int_0^{FOV_x} \int_0^{FOV_y} \rho(y, x) \cdot e^{ik_y y} \cdot e^{ik_x x} \cdot dy \cdot dx$$

$$k_x = \int_0^T G_x dt$$

$$k_y = \int_0^T G_y dt$$

Review: Spatial encoding in MRI

k-space principle:

$$S(k_y) = \int_0^{FOV_y} \rho(y) \cdot e^{ik_y y} dy$$

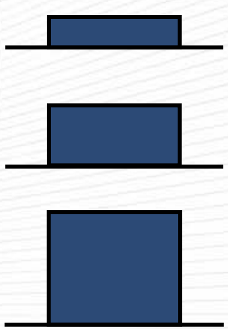
$$k_y = \int_0^T G_y dt$$

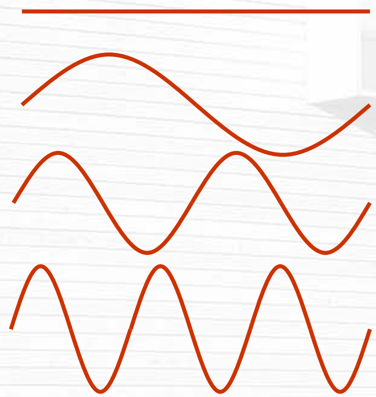
$S(k_y = 0 \cdot \Delta k_y)$ $\quad e^{i \cdot 0 \Delta k_y \cdot y}$

$S(k_y = 1 \cdot \Delta k_y)$ $\quad e^{i \cdot 1 \Delta k_y \cdot y}$

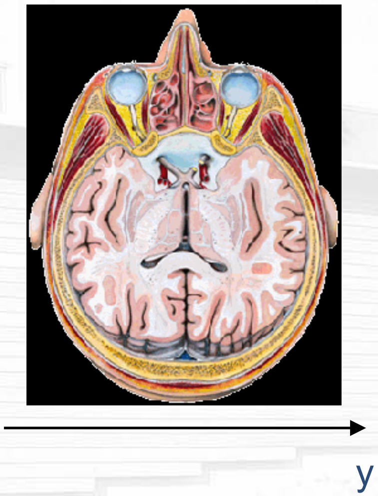
$S(k_y = 2 \cdot \Delta k_y)$ $\quad e^{i \cdot 2 \Delta k_y \cdot y}$

$S(k_y = 3 \cdot \Delta k_y)$ $\quad e^{i \cdot 3 \Delta k_y \cdot y}$





Object $\rho(y)$



Magnetic field gradients generate spatial harmonics!

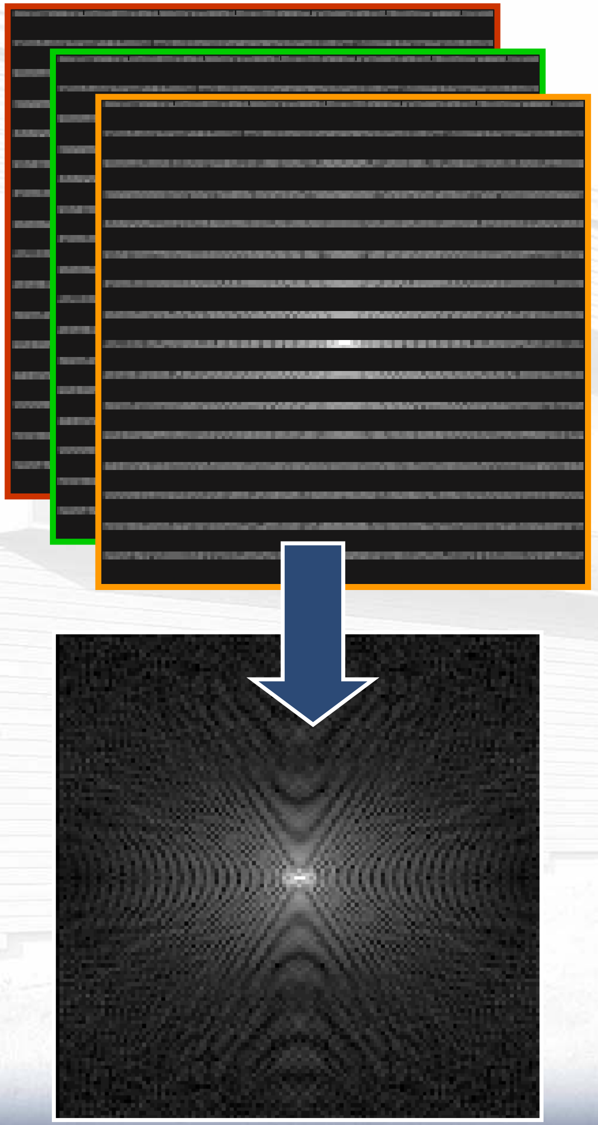
SMASH - The Math

Acquired lines:

$$S_l(k_y) = \int C_l(y) \cdot \rho(y) \cdot e^{ik_y y} dy$$

Not acquired lines:

$$S_l(k_y + m\Delta k_y) = \int C_l(y) \cdot \rho(y) \cdot e^{im\Delta k_y y} \cdot e^{ik_y y} dy$$



The difference between shifted k-space lines:
spatial harmonics of order m !!!

Is it possible ....?

$$S^{comp}(k_y + m\Delta k_y) = \sum_{l=1}^{N_C} w_l^{(m)} \cdot S_l(k_y) \qquad m = 0,\ldots, R-1$$

SMASH - The Math

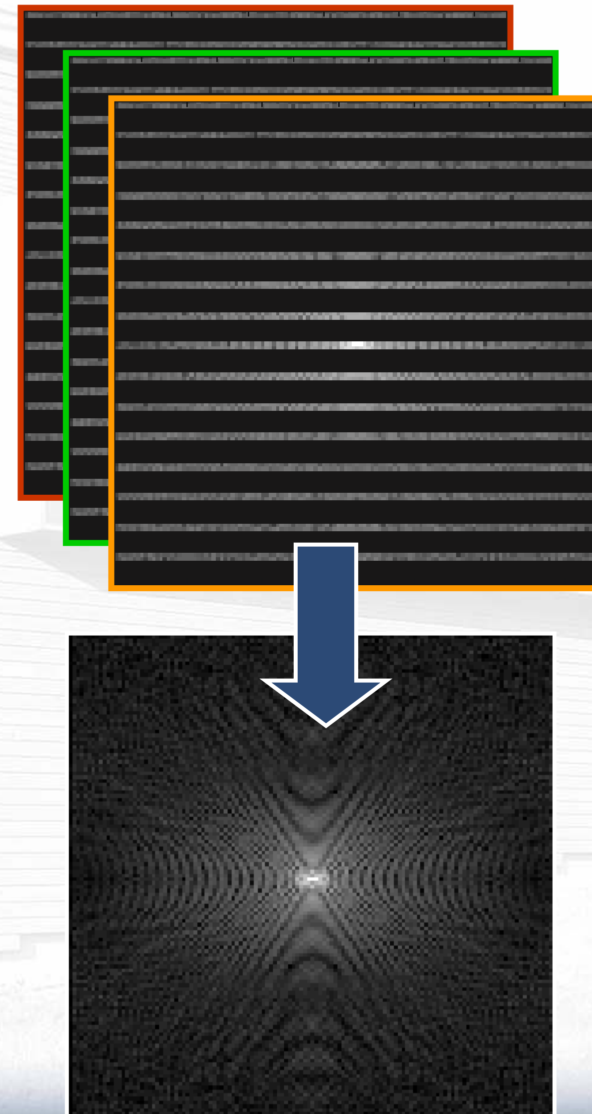
- **Is it possible ….?**

$$S^{Comp}(k_y + m\Delta k_y) = \sum_{l=1}^{N_C} w_l^{(m)} \int C_l(y) \cdot \rho(y) \cdot e^{ik_y y}\, dy$$

$$S^{Comp}(k_y + m\Delta k_y) = \int \sum_{k=1}^{N_C} w_l^{(m)} \cdot C_l(y) \cdot \rho(y) \cdot e^{ik_y y}\, dy$$

- **Yes, if….!**

$$\sum_{l=1}^{N_C} w_l^{(m)} \cdot C_l(y) \approx e^{im\Delta k_y y}$$

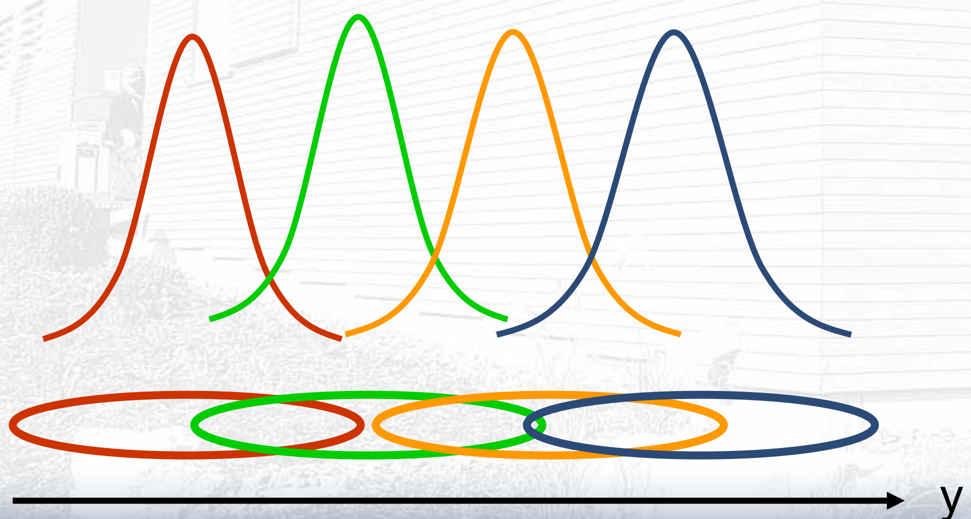


SMASH - The Picture

- **SMASH allows to synthesize missing k-space lines from undersampled multi-coil data……**

$$S^{Comp}(k_y + m\Delta k_y) = \sum_{k=1}^{N_C} w_k^{(m)} \cdot \tilde{S}_k(k_y) \qquad \sum_{k=1}^{N_C} w_k^{(m)} \cdot C_k(y) \approx e^{im\Delta k_y y}$$

- **…. if the coil profiles can be combined to built spatial harmonics of order m!**

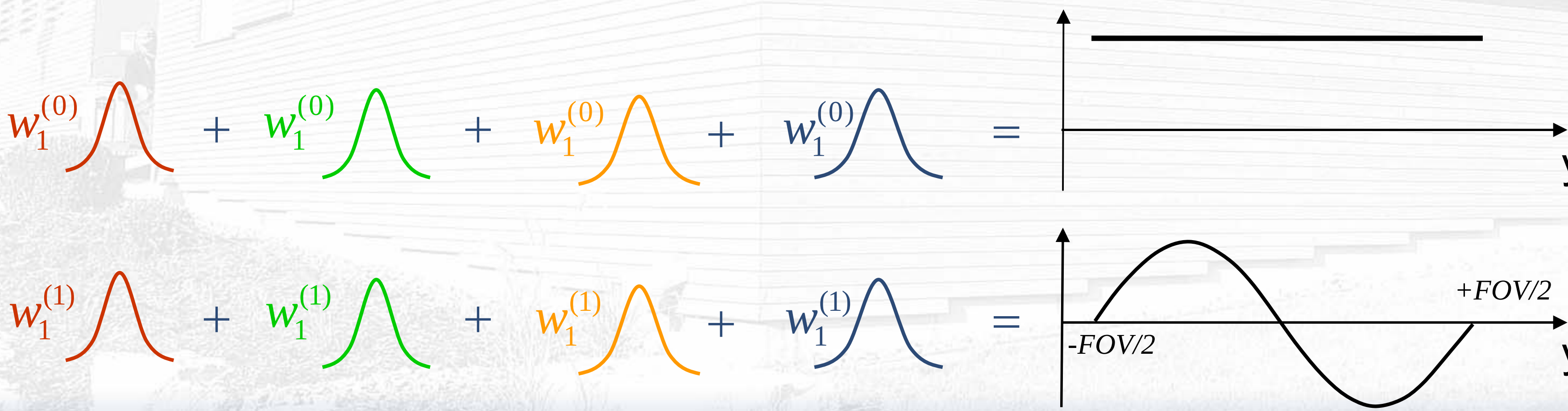


SMASH - The Picture

- SMASH allows to synthesize missing k-space lines from undersampled multi-coil data……

$$S^{Comp}(k_y + m\Delta k_y) = \sum_{k=1}^{N_C} w_k^{(m)} \cdot \tilde{S}_k(k_y) \qquad\qquad \sum_{k=1}^{N_C} w_k^{(m)} \cdot C_k(y) \approx e^{im\Delta k_y y}$$

- …. if the coil profiles can be combined to built spatial harmonics of order m!



SMASH - The Picture

- **SMASH allows to synthesize missing k-space lines from undersampled multi-coil data……**

$$S^{Comp}(k_y + m\Delta k_y) = \sum_{k=1}^{N_C} w_k^{(m)} \cdot S_k(k_y)$$

- **…. if the coil profiles can be combined to built spatial harmonics of order m!**

$$\sum_{k=1}^{N_C} w_k^{(m)} \cdot C_k(y) \approx e^{im\Delta k_y y} \qquad \vec{w}^{(m)} \cdot \hat{c} = \vec{h}^{(m)} \qquad \vec{w}^{(m)} = \vec{h}^{(m)} \cdot \mathrm{pinv}(\hat{c})$$

SMASH - Extensions

- **Conventional SMASH**

$$\sum_{k=1}^{N_C} w_k^{(m)} \cdot C_k(y) \approx e^{im\Delta k_y y}$$

$$S^{Comp}(k_y + m\Delta k_y) = \sum_{k=1}^{N_C} w_k^{(m)} \cdot S_k(k_y)$$

- **Tailored SMASH**

$$\sum_{k=1}^{N_C} w_k^{(m)} \cdot C_k(y) \approx e^{i(m-0.5)\Delta k_y y}$$

$$S^{Comp}(k_y + (m-0.5) \cdot \Delta k_y) = \sum_{k=1}^{N_C} w_k^{(m)} \cdot S_k(k_y)$$

$$\sum_{k=1}^{N_C} w_k^{(m)} \cdot C_k(y,x) \approx f(y,x) \cdot e^{im\Delta k_y y}$$

- **"Coil-by-Coil" SMASH (half the way to GRAPPA)**

$$\sum_{k=1}^{N_C} w_{lk}^{(m)} \cdot C_k(y,x) \approx C_l(y,x) \cdot e^{im\Delta k_y y}$$

$$S_l(k_y + m\Delta k_y) = \sum_{k=1}^{N_C} w_{lk}^{(m)} \cdot S_k(k_y)$$

→ Weight set for each coil: No composite signal →single coil signals

SMASH - The Code

- **Create spatial harmonics h of order m=0,1,2 … (af-1)**
  - y = [-ny/2+1:ny/2];
  - h(m,:) = exp(i*2*pi/ny*m*y);
- **Get coil profiles c from sensitivity maps**
  - c=cmap(:,:,x);
- **Solve for the SMASH weights at x using the pseudo inverse**
  - w(m)*c=h(m) ➔ w(m)=h(m)*pinv(c)
- **Apply weights to undersampled data and write result in matrix**
  - sig_reco(m+1:af:ny,x)=w(m)*sig_hbd(:,:,x);
- **FFT data in image space**
  - reco = ifftshift(ifft(ifftshift(sig_reco,1),[],1),1);
- **That's it….**

$$\sum_{k=1}^{N_C} w_k^{(m)} \cdot C_k(y) \approx e^{im\Delta k_y y}$$

$$\begin{pmatrix} w_1^{(m)} & w_2^{(m)} & w_3^{(m)} & w_4^{(m)} \end{pmatrix} \cdot \begin{pmatrix} C_1(y) \\ C_2(y) \\ C_3(y) \\ C_4(y) \end{pmatrix} \approx h^{(m)}(y)$$

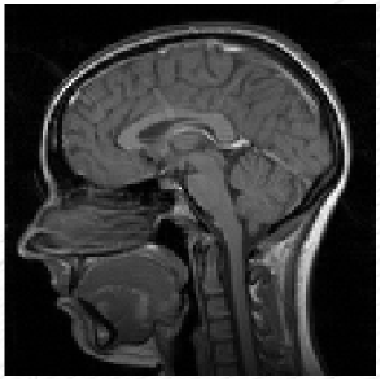
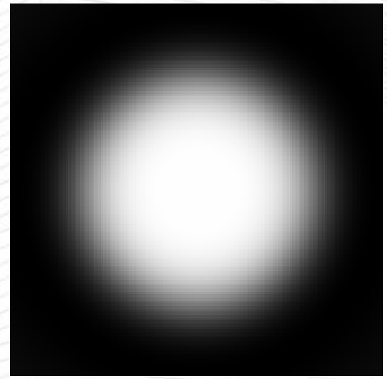
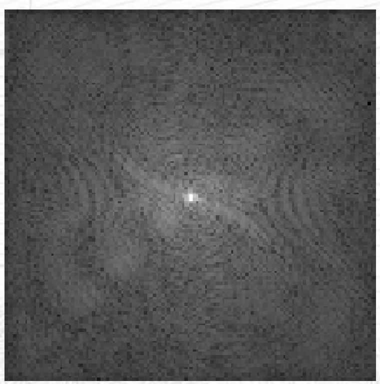
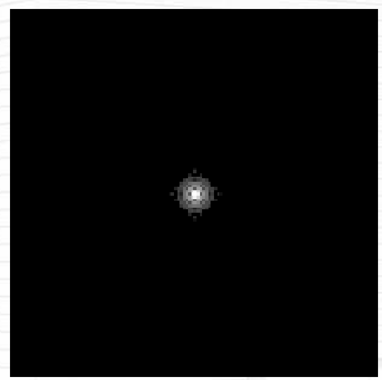
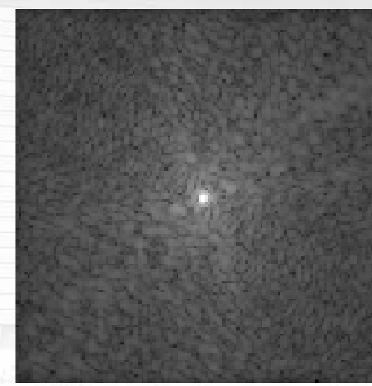
pMRI reconstructions in k-space

SMASH (Sensitivity maps)

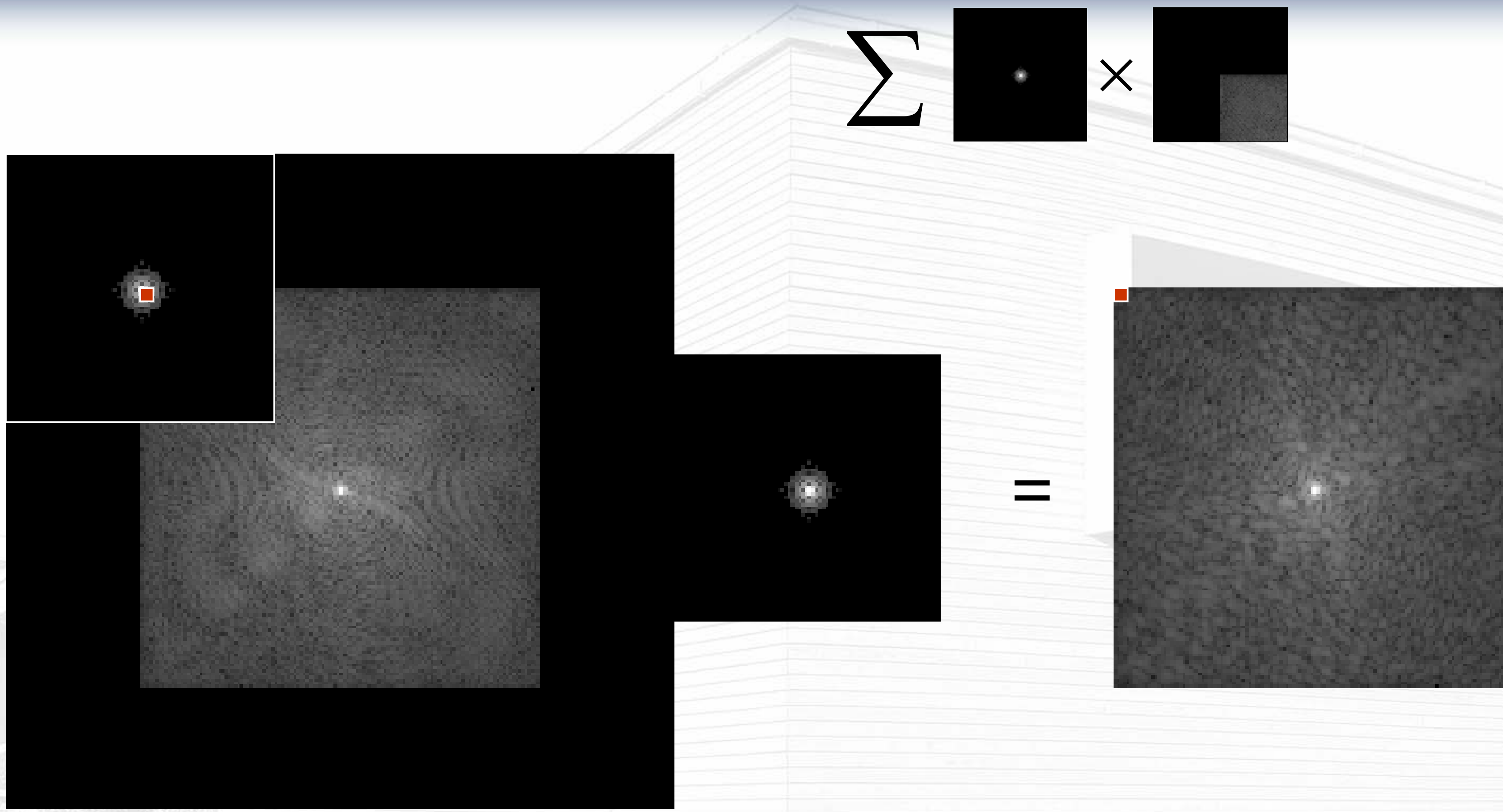
Generalized SMASH (Sensitivity maps)

GRAPPA (Calibration data set)

Review: Image and k-space domains

	object		coil sensitivity		coil image
Image Domain multiplication		$\times$		$=$	
					FT
k-space convolution		$\otimes$		$=$	
	object k-space		coil sensitivity k-space		coil image k-space

2D Convolution: How does it work?



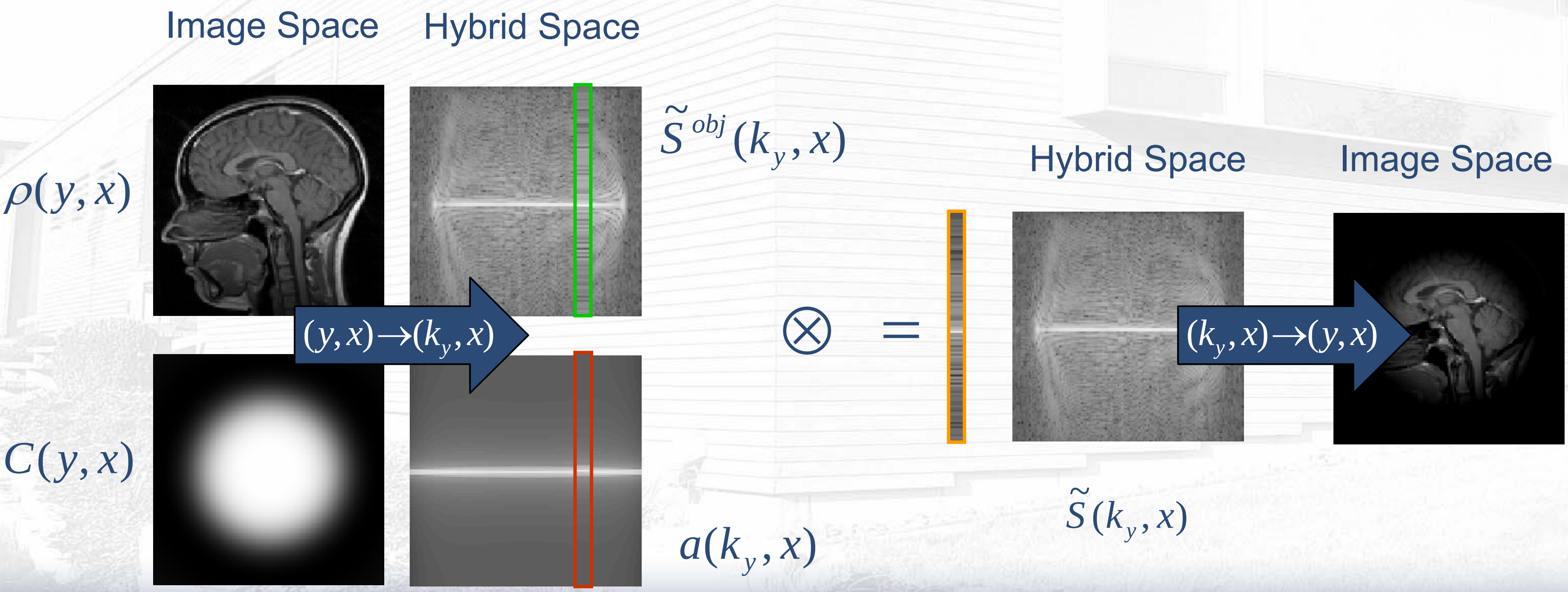
G-SMASH - Convolution in 1D

Fully encoded Hybrid Space: $(k_y, k_x) \rightarrow (k_y, x)$

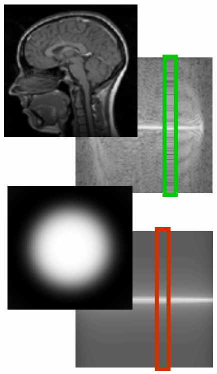
$$\tilde{S}(k_y, x) = \int C(y,x) \cdot \rho(y,x) \cdot e^{ik_y y}\, dy$$

$$\tilde{S}(k_y, x) = a(k_y, x) \otimes \tilde{S}^{obj}(k_y, x)$$

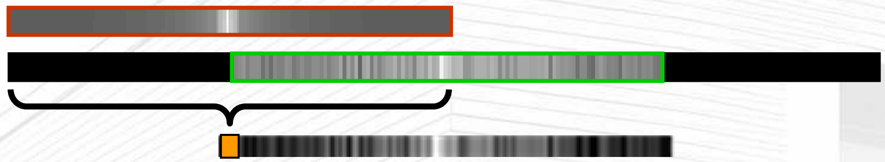
$$a(k_y, x) = \int C(y,x) \cdot e^{ik_y y}\, dy$$

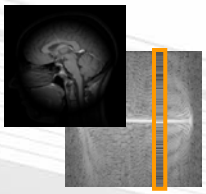


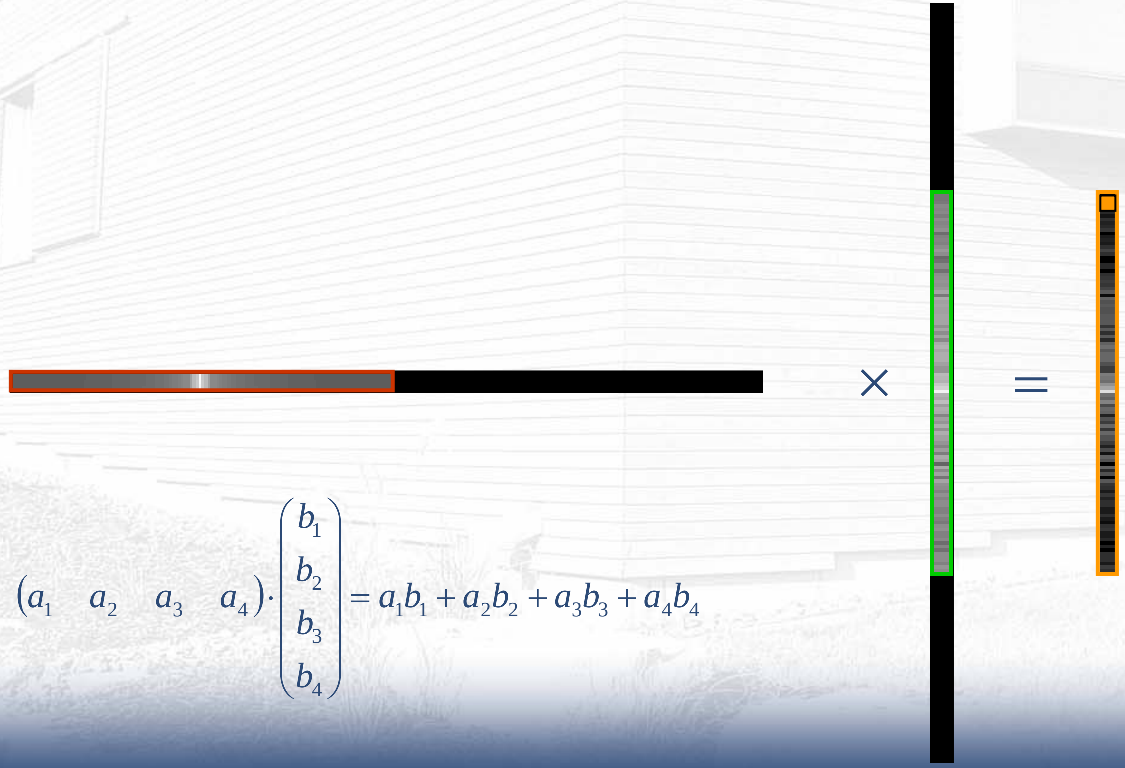
1D Convolution in Matrixform



$$\tilde{S}(k_y) = a(k_y) \otimes \tilde{S}^{obj}(k_y)$$



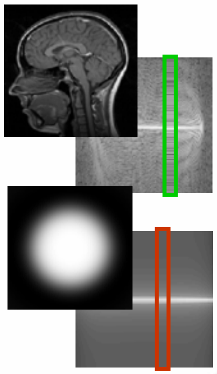


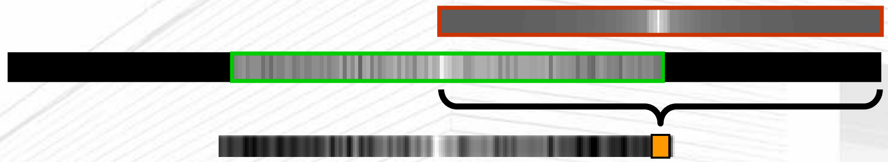


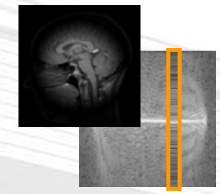
$$\begin{pmatrix} a_1 & a_2 & a_3 & a_4 \end{pmatrix} \cdot \begin{pmatrix} b_1 \\ b_2 \\ b_3 \\ b_4 \end{pmatrix} = a_1 b_1 + a_2 b_2 + a_3 b_3 + a_4 b_4$$

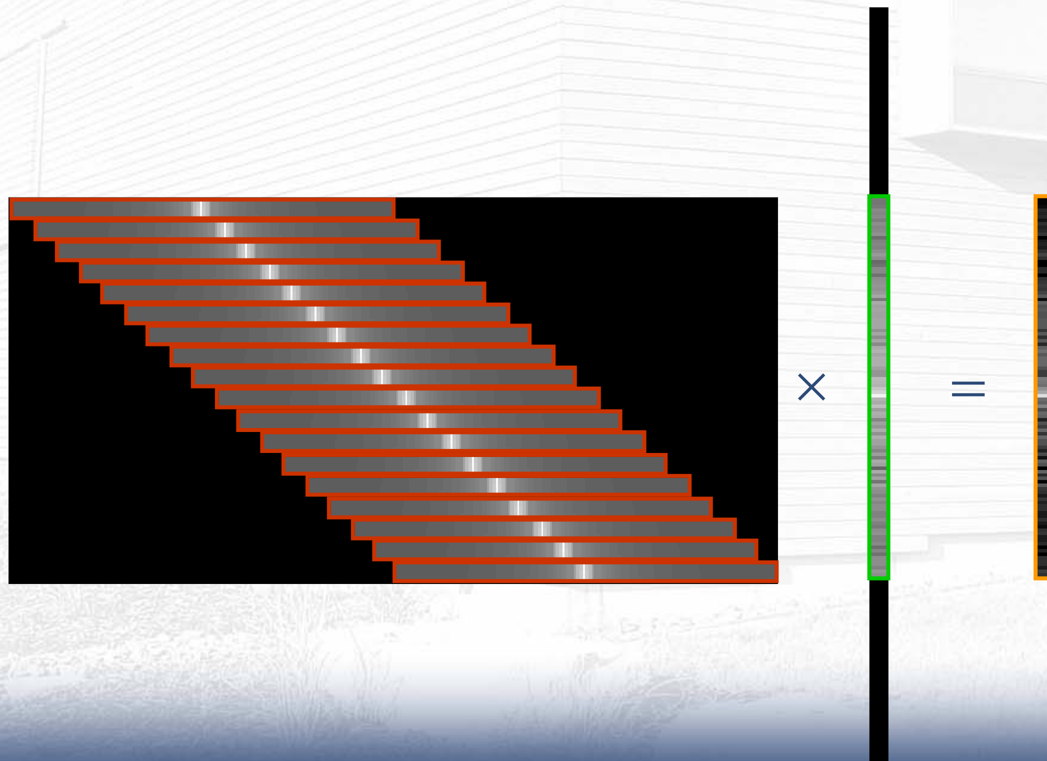
1D Convolution in Matrixform

$$\tilde{S}(k_y) = a(k_y) \otimes \tilde{S}^{obj}(k_y)$$



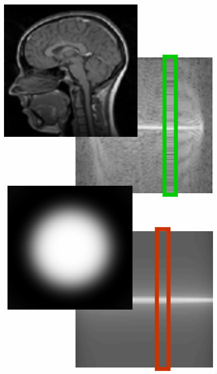


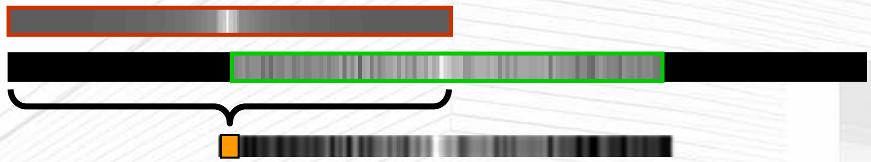


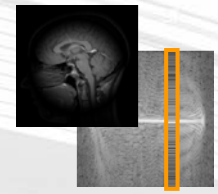


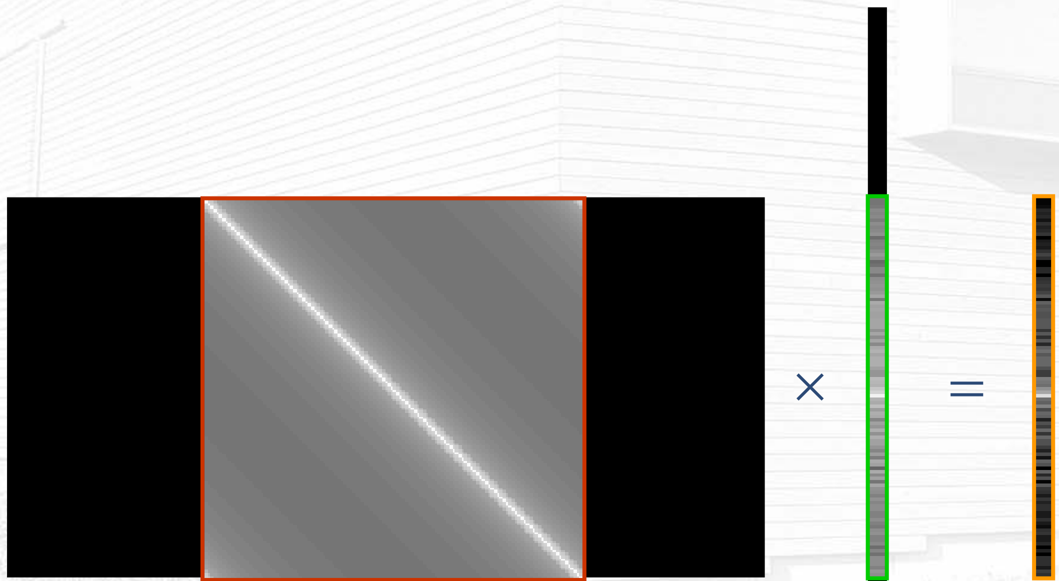
1D Convolution in Matrixform

$$\tilde{S}(k_y) = a(k_y) \otimes \tilde{S}^{obj}(k_y)$$







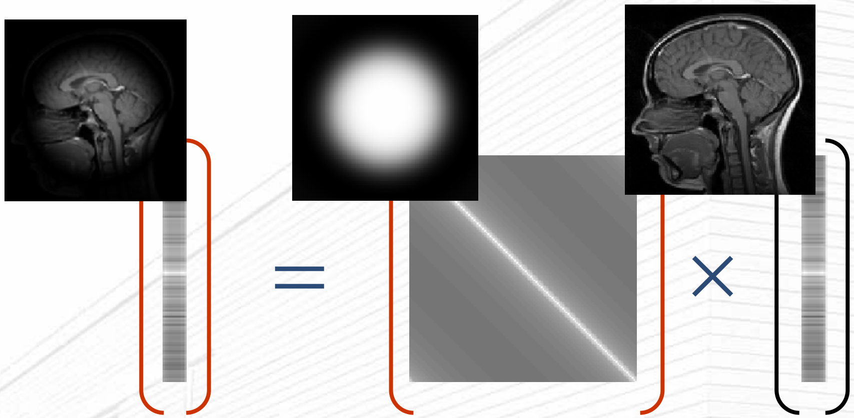


periodic nature of data, wrap around …

G-SMASH – The Math

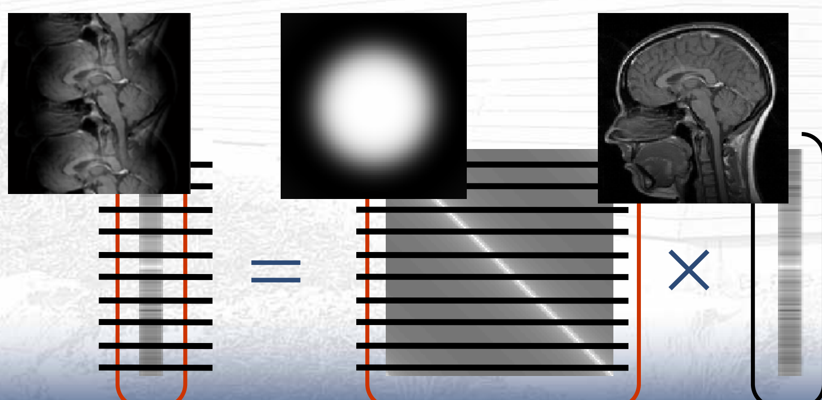
- **fully encoded data**

$$\tilde{S}(k_y) = a(k_y) \otimes \tilde{S}^{obj}(k_y)$$



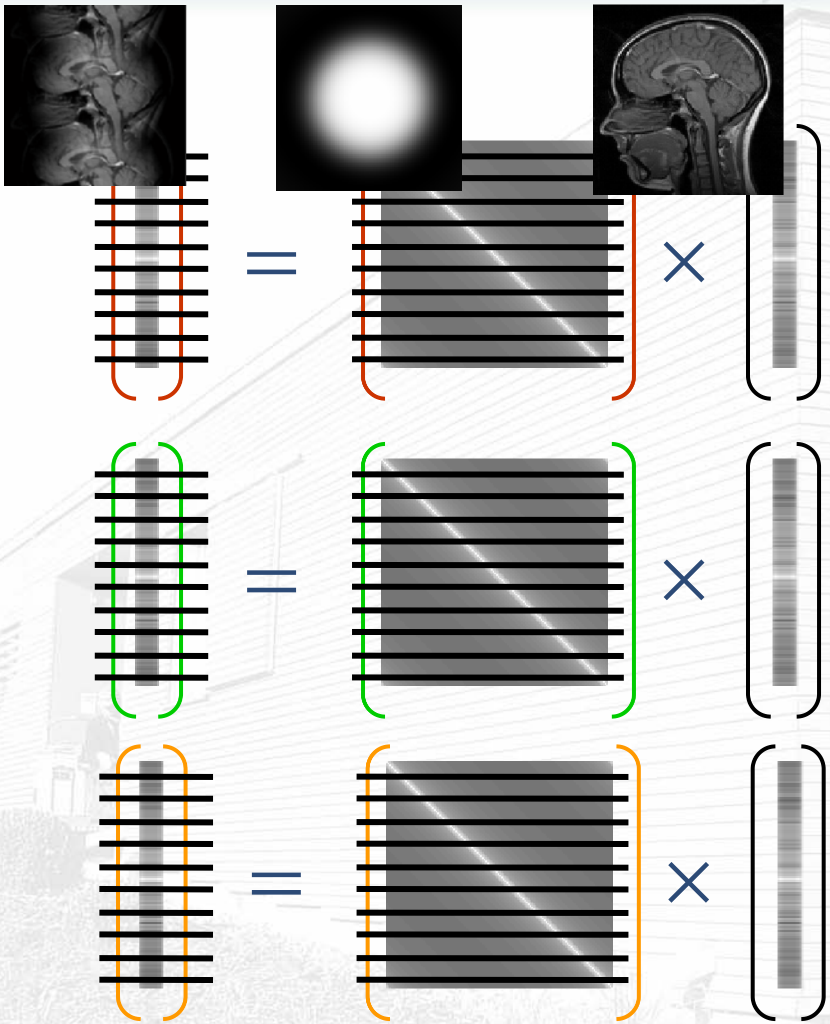
$$\vec{S}_l^{full} = \hat{C}_l^{full} \cdot \vec{S}^{obj}$$

- **(ir)regularly undersampled data**



$$\vec{S}_l^{red} = \hat{C}_l^{red} \cdot \vec{S}^{obj}$$

G-SMASH – The Math



$$\tilde{S}(k_y) = a(k_y) \otimes \tilde{S}^{obj}(k_y)$$

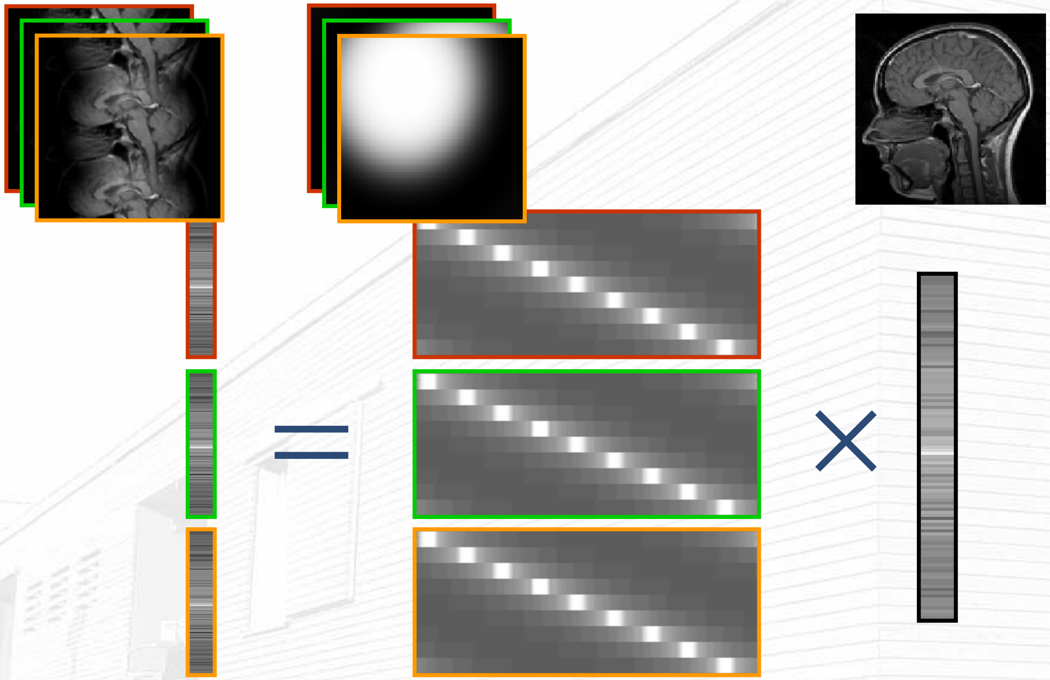
N/R equations, N unknowns

N/R equations, N unknowns

N/R equations, N unknowns

G-SMASH – The Math

$$\tilde{S}_l(k_y) = \tilde{C}_l(k_y) \otimes \tilde{\rho}(k_y)$$



N/R · #Coils equations, N unknowns

$$\vec{S}^{\,red}_{coils} = \hat{C}^{red}_{coils} \cdot \vec{S}^{\,full}_{obj}$$

$$\vec{S}^{\,full}_{obj} = \mathrm{pinv}\left(\hat{C}^{red}_{coils}\right) \cdot \vec{S}^{\,red}_{coils}$$

G – SMASH – The link to SMASH

- **SMASH**

$$\sum_{k=1}^{N_C} w_k^{(m)} \cdot C_k(y) \approx e^{im\Delta k_y y}$$

$$\tilde{S}^{Comp}(k_y + m\Delta k_y) = \sum_{k=1}^{N_C} w_k^{(m)} \cdot \tilde{S}_k(k_y)$$

- **Generalized - SMASH**

$$C_k(y) \approx \sum_{m=-N/2}^{N/2} a_k^{(m)}(x) \cdot e^{im\Delta k_y y}$$

$$\tilde{S}_k(k_y, x) = \sum_{m=-N/2}^{N/2} a_k^{(m)}(x) \int \rho(y,x) \cdot e^{im\Delta k_y y} \cdot e^{ik_y y} dy$$

$$\tilde{S}_k(k_y, x) = \sum_{m=-N/2}^{N/2} a_k^{(m)}(x) \cdot \tilde{S}^{obj}(k_y + m\Delta k_y, x)$$

G-SMASH - The Code

- **Transform undersampled k-space data into hybrid space**
  - sig_hbd = ifftshift(ifft(ifftshift(sig,2),[],2),2)
  - ky_red=ky/af
- **Transform coil sensitivity maps (y,x) into hybrid space (ky,x)**
  - cmap_hbd = fftshift(fft(fftshift(cmap,3),[],3),3)
- **Create Fourier coefficient matrix c at position x for every measured line and coil**
  - m=1,2,…ky_red
  - c(m,:,:) = circshift(squeeze(cmap_hbd(:,:,x)),[0 (ny/2-(m-1)*af)]);
  - reshape and reorder matrix to size (kyred*#coils x ky) and measured signal to (kyred*#Coils)
- **Solve for the fully encoded hybrid signal using the pseudo inverse**
  - sig_hbd=c*sig ➔ sig=pinv(c)*sig_hbd
- **Transform reconstructed hybrid signal (ky,x) in image space (y,x)**
  - reco = ifftshift(ifft(ifftshift(sig_reco,1),[],1),1);
  - Be careful, data might need to be flipped and shifted by one pixel
- **That's it….**

Generalized SMASH

- **Pros**
  - No special coil configuration required (= SENSE, ≠ SMASH!)
  - Less sensitive to coil map errors (coil information is transformed into k-space)
  - Works for non-uniform undersampling schemes

- **Cons**
  - Requires explicit sensitivity maps (= SENSE)
  - Reconstruction is relatively slow (big matrix to invert)

pMRI reconstructions in k-space

SMASH (Sensitivity maps)

Generalized SMASH (Sensitivity maps)

GRAPPA (calibration scan)

GRAPPA-The Math

- **We know from SMASH**

$$S(k_y + m\Delta k_y) = \sum_{l=1}^{N_c} w_l^{(m)} \cdot S_l(k_y)$$

$$\sum_{l=1}^{N} w_l^{(m)} C_l(y,x) \approx e^{im\Delta k_y y}$$

- **We know from “Coil by Coil” – SMASH**

$$S_k(k_y + m\Delta k_y) = \sum_{l=1}^{N_c} w_{kl}^{(m)} \cdot S_l(k_y)$$

$$\sum_{l=1}^{N} w_{kl}^{(m)} C_l(y,x) \approx C_k(y,x) \cdot e^{im\Delta k_y y}$$

- **Can the weights be spatially dependent…..?**

$$w_{kl}^{(m)} \to W_{kl}^{(m)}(y,x)$$

$$\sum_{l=1}^{N} W_{kl}^{(m)}(y,x) \cdot C_l(y,x) \approx C_k(y,x) \cdot e^{im\Delta k_y y}$$

GRAPPA – The MATH

- **What if …..**

$$S_k(k_y + m\Delta k_y, k_x) = \sum_{l=1}^{N_c} \iint W_{kl}^{(m)}(y,x) \cdot C_l(y,x) \cdot \rho(y,x) \cdot e^{ik_y y} \cdot e^{ik_x x} dy \cdot dx$$

- **Convolution Theorem…..**

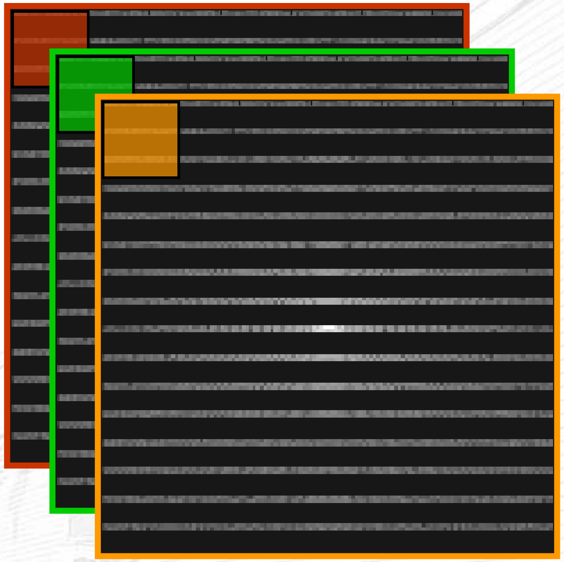
$$S_k(k_y + m\Delta k_y, k_x) = \sum_{l=1}^{N_c} w_{kl}^{(m)}(k_y, k_x) \otimes S_l(k_y, k_x)$$

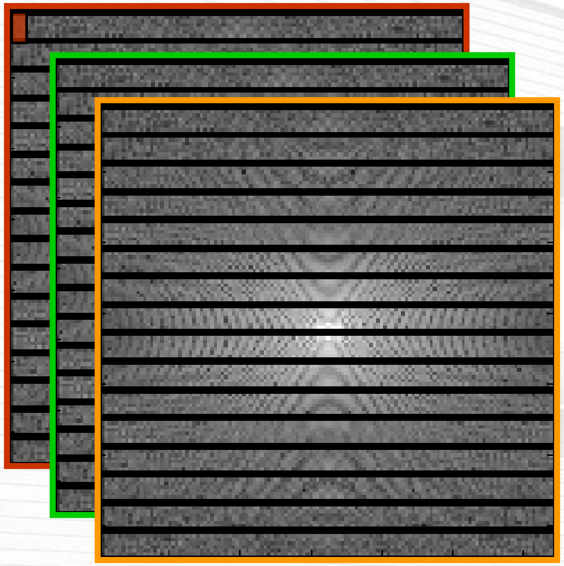
$$w_{kl}^{(m)}(k_y, k_x) \xleftrightarrow{\text{2D FFT}} W_{kl}^{(m)}(y,x)$$

$$S_l(k_y, k_x) \xleftrightarrow{\text{2D FFT}} C_l(y,x) \cdot \rho(y,x)$$

GRAPPA – The Picture

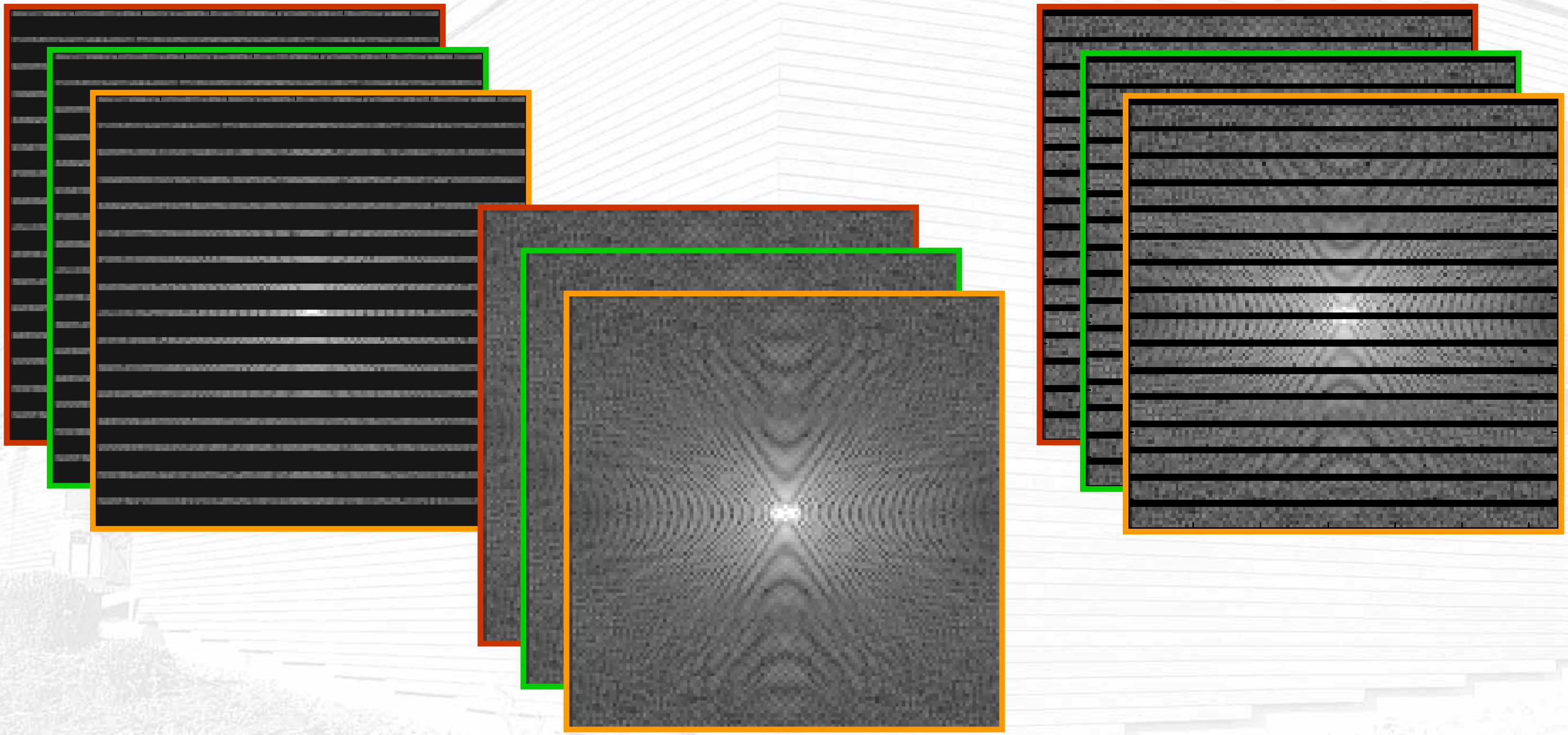
$$S_k(k_y + m\Delta k_y, k_x) = \sum_{l=1}^{N_c} w_{kl}^{(m)}(k_y, k_x) \otimes S_l(k_y, k_x)$$





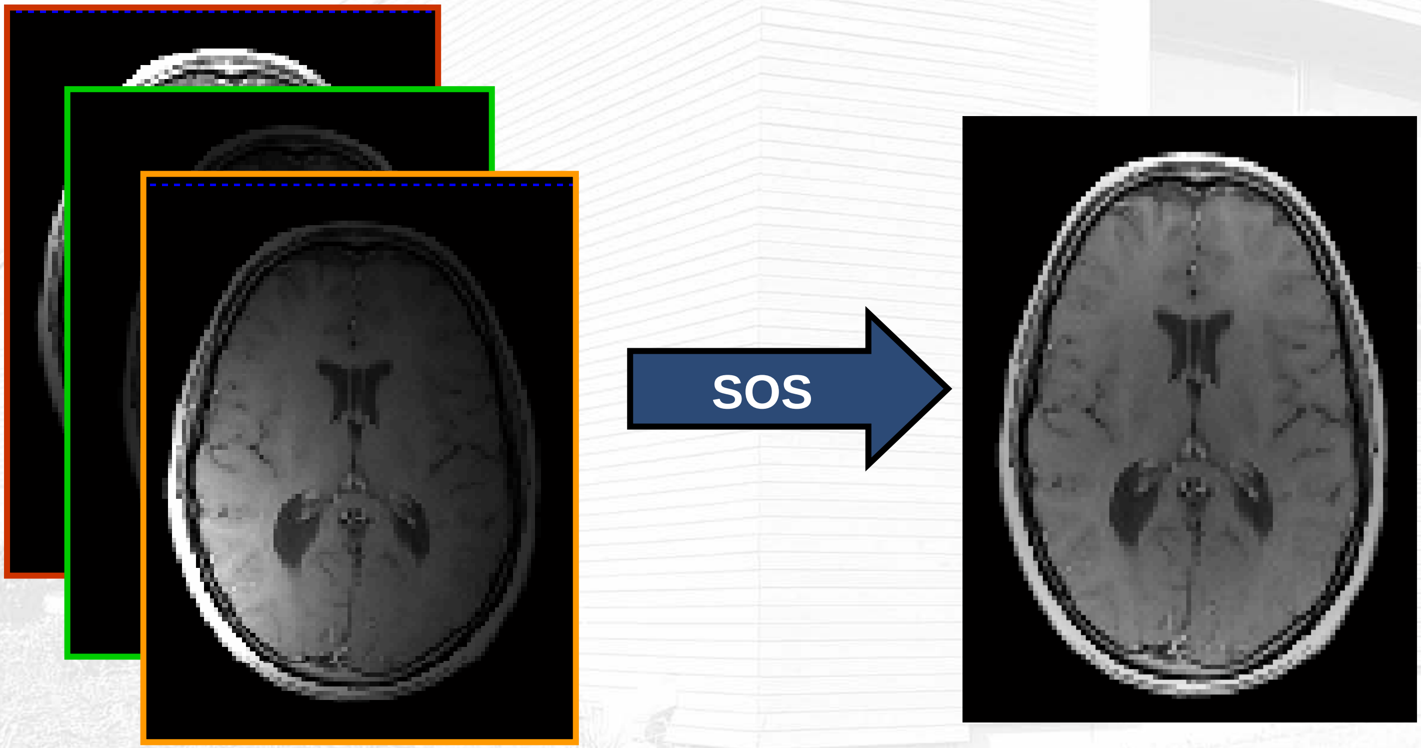
GRAPPA – The Picture

$$S_k(k_y + m\Delta k_y, k_x) = \sum_{l=1}^{N_c} w_{kl}^{(m)}(k_y, k_x) \otimes S_l(k_y, k_x)$$

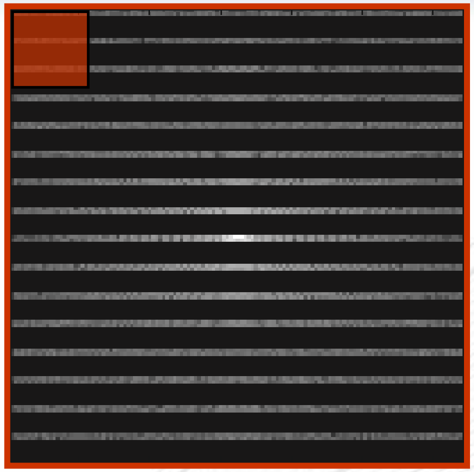


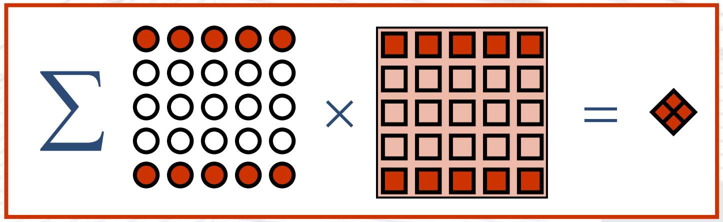
GRAPPA – The Picture

$$S_k(k_y + m\Delta k_y, k_x) = \sum_{l=1}^{N_c} w_{kl}^{(m)}(k_y, k_x) \otimes S_l(k_y, k_x)$$

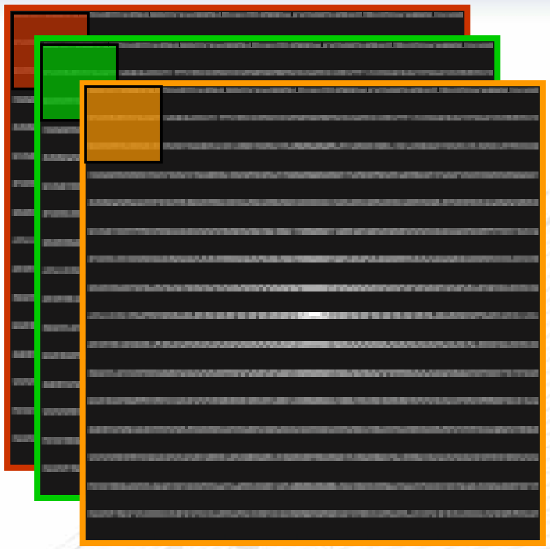


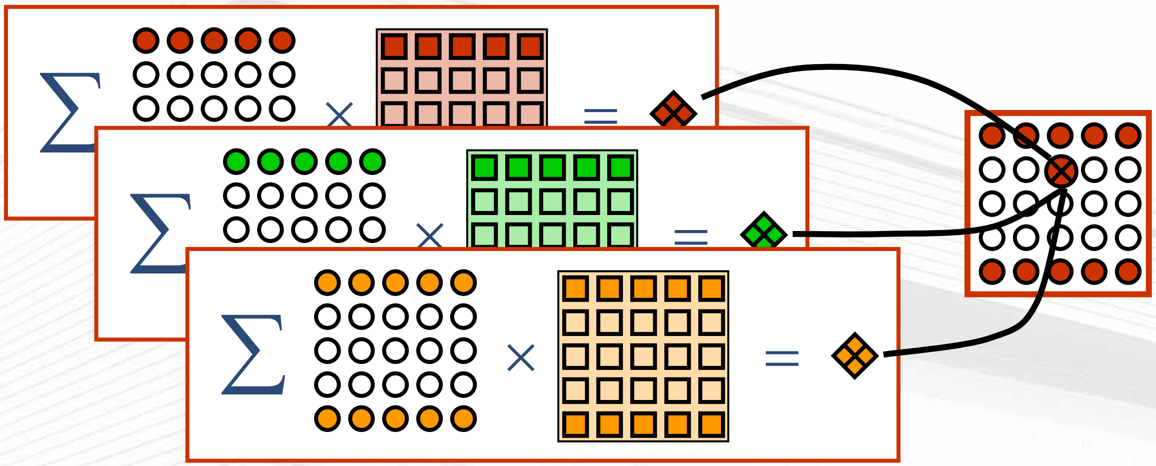
GRAPPA – A Closer Look





GRAPPA – A Closer Look

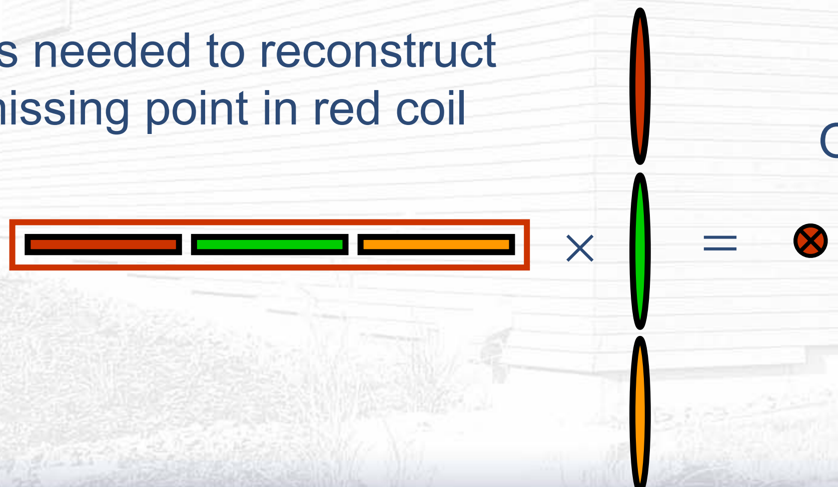




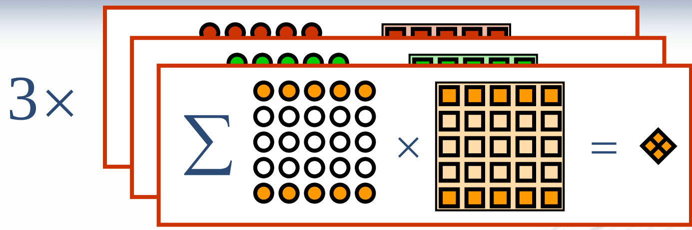
Acquired Signals in all coils in kernel region

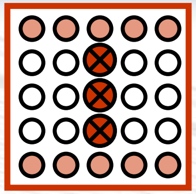
Weights needed to reconstruct one missing point in red coil

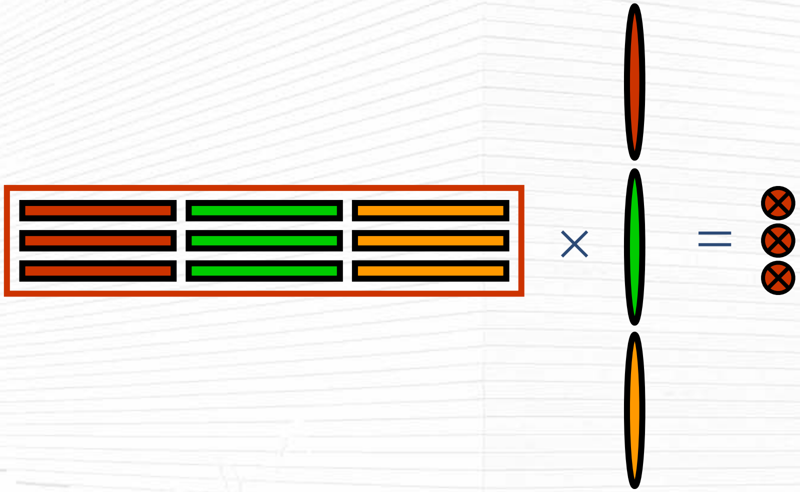
One missing point in red coil



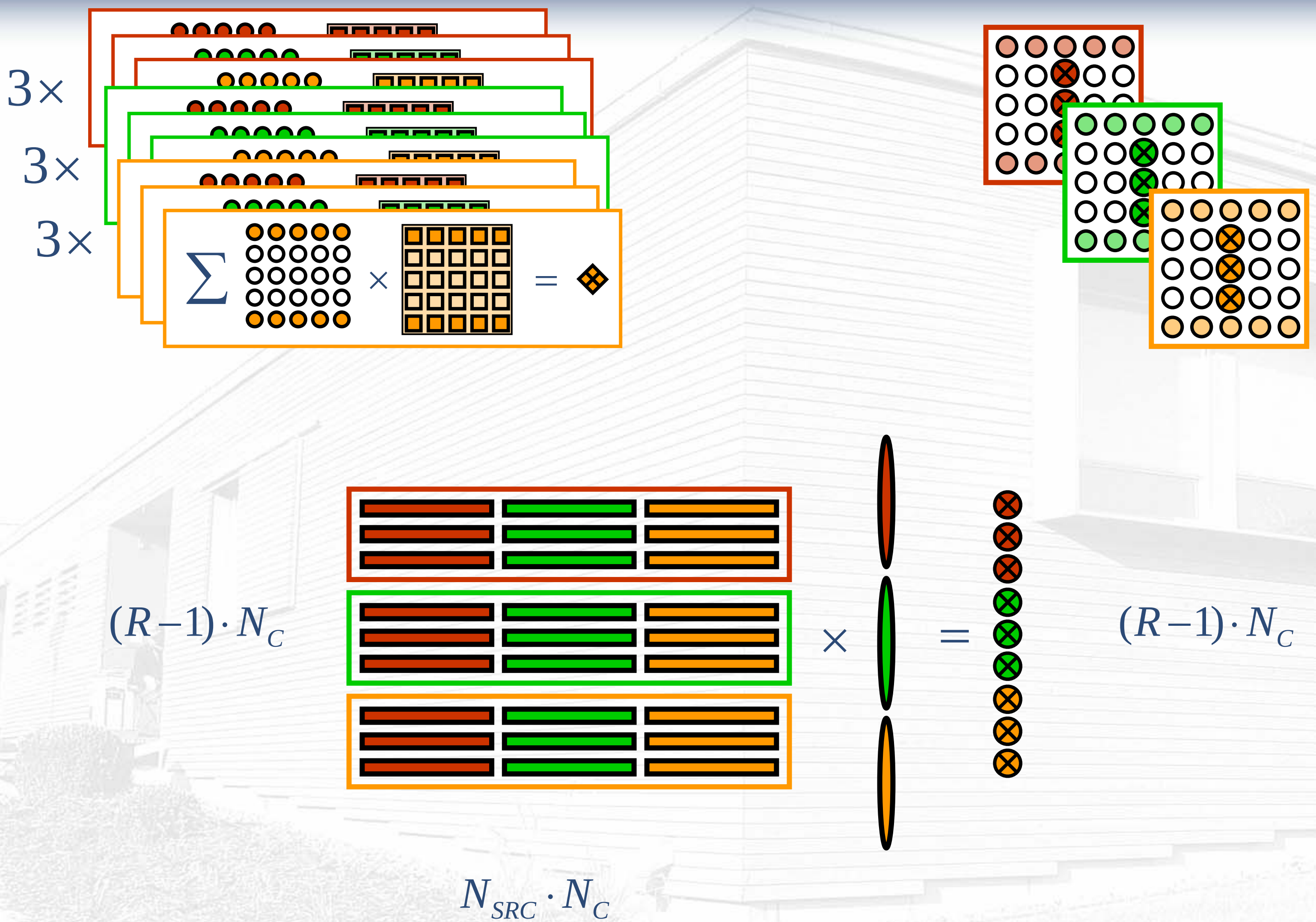
GRAPPA – A Closer Look



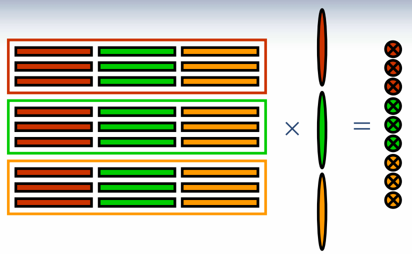




GRAPPA – A Closer Look



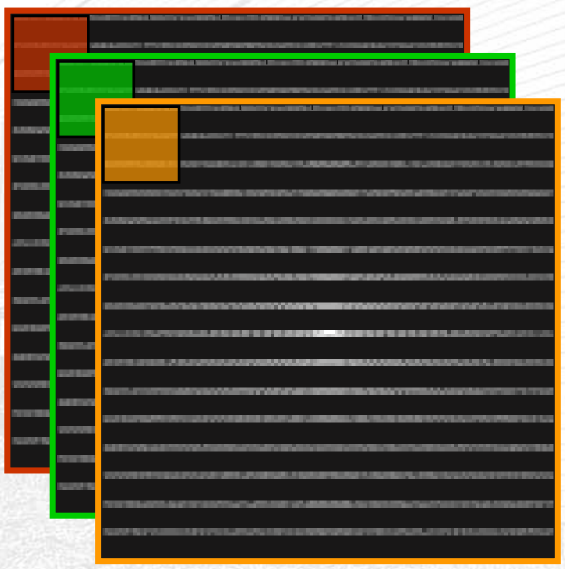
GRAPPA – A Closer Look

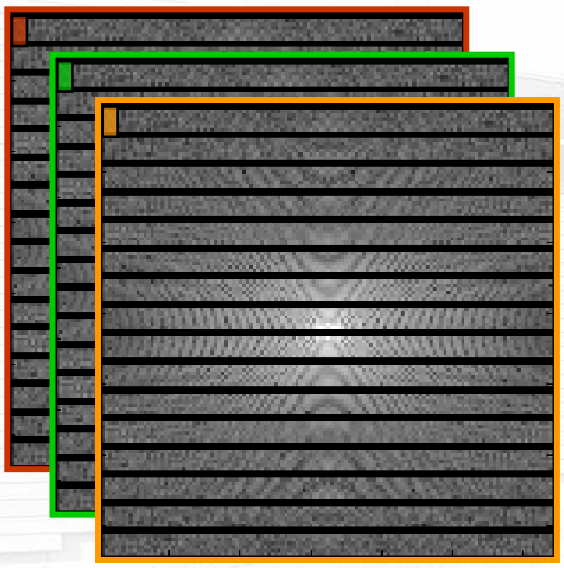


$\vec{S}_{REQ}$

$$\hat{W} \cdot \vec{S}_{ACQ} = \vec{S}_{REQ}$$

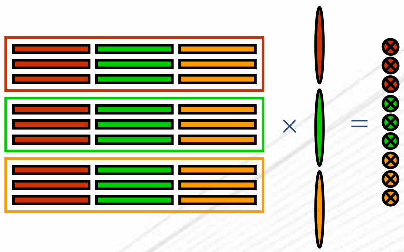
Repeat for every ky,kx





GRAPPA- How do we get the Weights

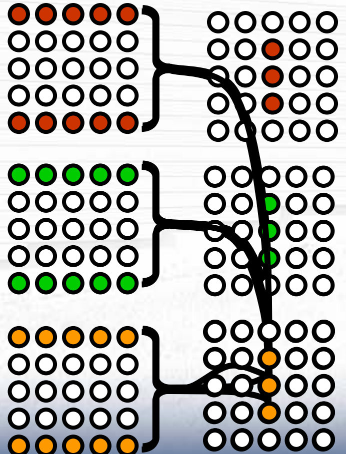
- **What we know is ….**



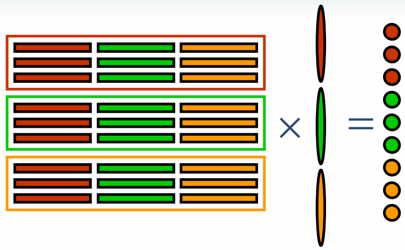
$$\hat{W}\cdot\vec{S}_{ACQ}=\vec{S}_{REQ}$$

- **Low Resolution ACS Dataset (eg. 32x32)**
  - can be acquired separately with arbitrary contrast
  - can be acquired during the scan (fully sampled center) and included in final image

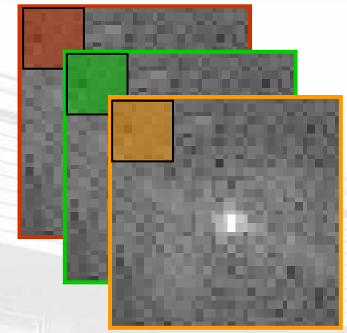
$$\hat{W}\cdot\vec{S}^{(ACQ)}_{ACS}=\vec{S}^{(REQ)}_{ACS}$$

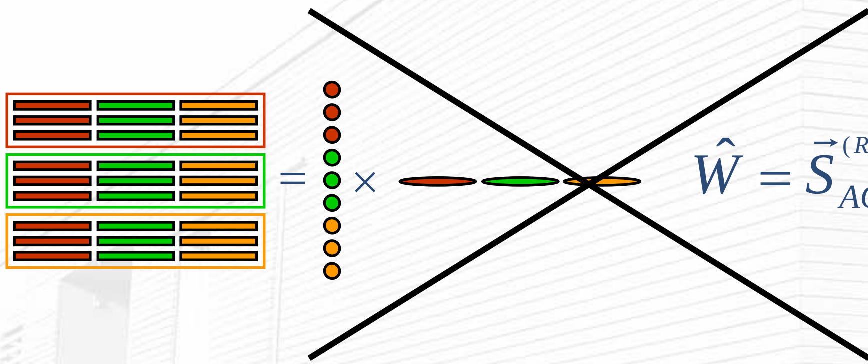


GRAPPA



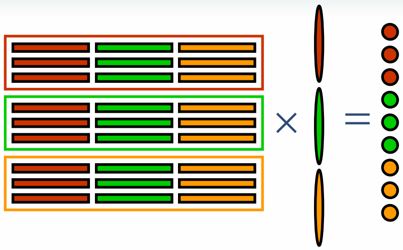
$$\hat{W} \cdot \vec{S}^{(ACQ)}_{ACS} = \vec{S}^{(REQ)}_{ACS}$$



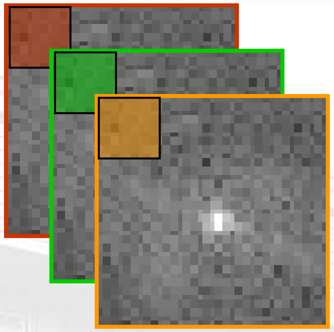


$$\hat{W} = \vec{S}^{(REQ)}_{ACS} \cdot \mathrm{pinv}\, \vec{S}^{(ACQ)}_{ACS}$$

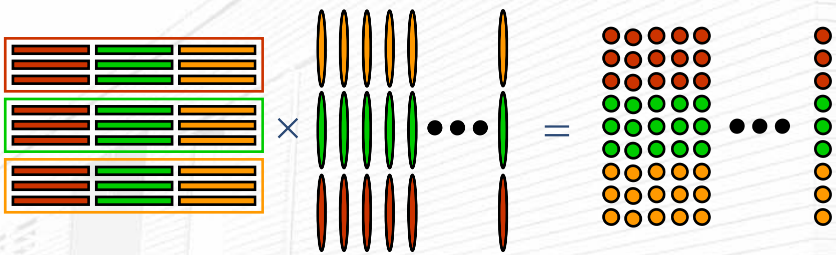
GRAPPA



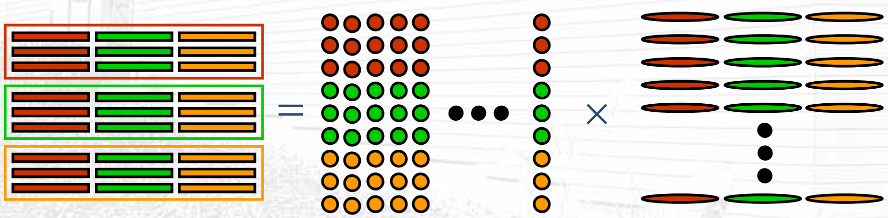
$$\hat{W}\cdot\vec{S}_{ACS}^{(ACQ)} = \vec{S}_{ACS}^{(REQ)}$$



Weights are the same, everywhere in k-space…..



$$\hat{W}\cdot\hat{S}_{ACS}^{(ACQ)} = \hat{S}_{ACS}^{(REQ)}$$



$$\hat{W} = \hat{S}_{ACS}^{(REQ)}\cdot \mathrm{pinv}\,\hat{S}_{ACS}^{(ACQ)}$$

GRAPPA – The Code

- **Weights Determination**
  - Choose GRAPPA kernel-size (eg, 4x5)
  - Choose ACS data size (eg, 32x32)
  - Assemble all source points in kernel region into a src matrix (#coils*4*5 x N)
  - Assemble all target points in kernel region into a trg matrix (#coils*(R-1) x N) (for all N repetitions in the ACS dataset)
  - Solve for the GRAPPA weights using pseudo inversion
    - w=trg*pinv(src);

- **Reconstruction at ky,kx**
  - Assemble all acquired signals in kernel region into vector sig_red (#coils*4*5)
  - Apply weights to data:
    - sig_reco = w*sig_red;
  - Reorder and reshape reconstructed data into final matrix:
    - sig_out(:,:,ky,kx) = reshape(sig_reco,[nc, R-1]);

- **Done …**

GRAPPA

- **„Coil by Coil“ Reconstruction**
  - No special coil configuration required (=SENSE, ≠ SMASH!)
  - Single coil images (≠ SENSE, SMASH and G-SMASH)
  - Almost optimal SNR

- **Autocalibration**
  - No explicit sensitivity maps required → ACS data
  - ACS can be included (→ effective reduction factor)
  - Very robust (even in the case of low signal, motion)
  - Fast

Acknowledgements

- Mark Griswold
- Peter Jakob
- Martin Blaimer
- Nicole Seiberlich
- David Larkman
- David Atkinson